BATUD: Blind Atmospheric TUrbulence Deconvolution

Charles-Alban Deledalle, Jérôme Gilles

BATUD: Blind Atmospheric TUrbulence Deconvolution

Charles Deledalle[1], *Jérôme Gilles*[2*]

[1] *Electrical and Computer Engineering Department, University of California San Diego, 9500 Gilman Dr, La Jolla, CA 92093, USA*
[2] *Department of Mathematics & Statistics, San Diego State University, 5500 Campanile Dr, San Diego, CA, 92182, USA*
* *E-mail: jgilles@sdsu.edu*

This paper is a preprint of a paper submitted to IET Image Processing. If accepted, the copy of record will be available at the IET Digital Library.

Abstract: A new blind image deconvolution technique is developed for atmospheric turbulence deblurring. The originality of the proposed approach relies on an actual physical model, known as the Fried kernel, that quantifies the impact of the atmospheric turbulence on the optical resolution of images. While the original expression of the Fried kernel can seem cumbersome at first sight, we show that it can be reparameterized in a much simpler form. This simple expression allows us to efficiently embed this kernel in the proposed Blind Atmospheric TUrbulence Deconvolution (BATUD) algorithm. BATUD is an iterative algorithm that alternately performs deconvolution and estimates the Fried kernel by jointly relying on a Gaussian Mixture Model prior of natural image patches and controlling for the square Euclidean norm of the Fried kernel. Numerical experiments show that our proposed blind deconvolution algorithm behaves well in different simulated turbulence scenarios, as well as on real images. Not only BATUD outperforms state-of-the-art approaches used in atmospheric turbulence deconvolution in terms of image quality metrics, but is also faster.

1 Introduction

Restoration of long range images acquired through the atmospheric turbulence has gain a lot of interest these last two decades. In astronomy, a preferred solution is to use adaptive deformable mirrors which try to compensate distortions created by the atmosphere. If such systems work reasonably well for astronomical images, they are very expansive, fragile and of large dimension making them unusable for ground-to-ground or air-to-ground applications, and hard to embed in terrestrial or aerial vehicles. An alternative approach is to use image processing techniques to mitigate the atmospheric turbulence. Such methods were widely investigated these last two decades. If a fluid mechanics point of view seems a natural way to consider the turbulence, the corresponding equations are too complicated to be used within an image processing framework. Instead, Frakes et al. [1] proposed to use a general operator point of view of the problem (usually not based on physics models). If we denote x the static original scene we want to recover, v_i the observed image at time i, D_i some geometric distortions due to the scattering effects of the turbulent atmosphere at time i and M a blurring kernel, a general atmospheric turbulence model impacting the observations can be written as

$$v_i = D_i(M(x)) + \text{noise}. \tag{1}$$

The blur M is commonly considered stationary (i.e it does not change on a short period of time) compared to the geometric distortions. The authors of [1] then use some adaptive grid interpolation technique to reconstruct a clean image from the input sequence (see also [2] for a more efficient approach). In [3], the authors followed the same idea but replace the interpolation process with a combination of elastic registration and low-rank filtering. In [4], the author proposed to use some local filtering process by block partitioning the image, however some block artifacts appear in the restored image. A system point of view is adopted in [5] where a Kalman filter is used to stabilize the images, followed by a Nonlocal Total Variation [6] deconvolution step to remove the blur. Several statistical arguments were use in the literature to mitigate atmospheric turbulence. In [7] and [8], the authors develop a centroid based approach, while (assuming long exposure video capture), Principal Component Analysis is used in [9]; an objective function combining temporal and spatial information is optimized by the SURE (Stein's Unbiased Risk Estimator) method in [10]. A method called "*lucky-region/imaging*" approach was proposed in [11], it is based on the assumption that, at specific locations and time, a neighborhood of good quality will appear with high probability. The method then aims at fusing the highest quality neighborhoods to recreate a clean image. If this method is originally motivated by the optical behavior of the atmosphere, its major drawback is the question of how to measure the quality of a neighborhood. In [12], Yang et al. proposed to combine some non-rigid registration and the above lucky-region method. Gilles et al. in [13], seek to estimate the inverse operators M^{-1} and D_i^{-1} by using some blind-deconvolution algorithm after performing a diffeomorphic registration step. Based on the same strategy, Zhu et al. in [14] used a B-Spline registration within a Bayesian framework, involving a bilateral total variation regularization, to estimate the inverse operators. A two steps method using first a multiscale optical flow estimation and then the FRTAAS algorithm (First Register Then Average And Subtract) was proposed in [15, 16] to obtained a restored image. The authors of [17] and [18] respectively use a generalized regression neural network and a convolutional neural network to learn turbulence induced deformations. These neural networks are then used to predict and compensate the turbulence impact. A fusion technique in a dual-tree complex wavelet domain combined with a non-rigid registration algorithm is used in [19]. This approach has two main drawbacks: 1) it requires a human intervention to select informative regions of interest, and 2) a segmentation step, whose efficacy varies, is required in the fusion strategy. In [20], the authors designed a multi-step algorithm which firstly estimates a reference frame via low-rank approximation; secondly, a variational model is used to enhance the quality of that reference frame. Next, this enhanced reference is used to estimate the deformation fields in the sequence which are then compensated to obtain a registered sequence. Finally a deformation guided fusion combined with a semi-blind deconvolution provides the restored image. In [21], Lau et al. proposed to minimize a functional, using different regularization terms, to perform specific subsampling of frames without any registration. A procedure based on optimal transport has been proposed in [22]. Several similar methods also use the strategy of combining a registration step with a deconvolution step, see for instance [23–27].

Some articles in the literature focus more specifically on either the geometric distortion compensation or the deblurring. For instance, in

[28, 29], the authors proposed a variational model combining optical flow and a Nonlocal Total Variation regularization term while, in [30], the authors used a rapid control grid optical flow estimation to get rid of the geometric distortions. A comparison of several registration techniques applied to turbulence distortions can be found in [31]. The deblurring aspect has been considered by Hirsch et al. in [32] where the authors used a spatially variant deconvolution technique. Li et al. in [33] used a Wiener filtering approach with a blur model whose parameters are estimated by minimizing the kurtosis of the restored image. A combination of Sobolev gradients and Laplacian in a unified framework applied on a previously stabilized image was proposed in [34]. Inspired by the lucky-imaging principle, a wavelet burst technique was proposed in [35] to deblur a sequence of stabilized images.

All these techniques have their specific pros and cons where generally most efficient methods, in terms of restoration quality, are the most computationally expansive. It is then a tradeoff between quality and computation which drive the user to choose one method or the other given the specific constraints for his application. As mentioned earlier, none of the methods listed above use any knowledge coming from a physical model. However, in 1966, Fried [36] investigated the physics involved on how the atmospheric turbulence impacts the optical resolution of images. His work was revisited by Tofsted [37] who provided a modern version of this model. The main result of this work is an analytical expression of the modulation transfer function (MTF), i.e the convolution kernel, corresponding to the impact of the atmospheric turbulence. In [38], Gilles et al. investigated the use of this MTF in a framelet based deconvolution algorithm. If the deblurring results are superior than the ones obtained by "generic" deconvolution algorithms, the proposed method has a major drawback: the expression of this kernel depends on four physical parameters. If it is reasonable to assume that three of them can be known/measured, the fourth one, called C_n^2 and representing how strong the turbulence is, is less obvious to know. In [38], the authors proposed some sort of "brute force" approach to estimate this parameter but it is computationally intensive since many deconvolutions must be processed. In this paper, we revisit this deconvolution approach and propose a simplified version of the Fried kernel depending only into two parameters which can be estimated numerically in an efficient way using the proposed BATUD (Blind Atmospheric TUrbulence Deconvolution) algorithm.

The rest of this paper is organized as follows. In Section 2, we recall the original analytical form of the Fried kernel. We proposed our simplified expression of the Fried kernel as well as the BATUD algorithm in Section 3. In Section 4, we present numerous experiments on both simulated and real images, and show the superiority of the proposed approach against several standard blind deconvolution algorithms. We finally conclude this work in Section 5.

2 Fried kernel

The Fried kernel was first proposed in [36] and then revisited in [37]. In this paper, we use the formalism of the latter article. The Fried kernel can be viewed as a combination of two terms. The first one, denoted M_0, corresponds to a combination of the system plus atmosphere MTFs when the turbulence is negligible. The second term, denoted M_{SA} and also called the short-term exposure MTF, models the impact (in term of blur) of phase-tilt due to the turbulence. Denoting ω the spatial frequency (in 2D we consider an isotropic kernel and ω is the frequency modulus), $M_0(\omega)$ can be expressed by

$$M_0(\omega) = \begin{cases} \frac{2}{\pi}\left(\arccos(|\omega|) - |\omega|\sqrt{1-|\omega|^2}\right) & |\omega| < 1 \\ 0 & |\omega| > 1, \end{cases} \tag{2}$$

and $M_{SA}(\omega)$ is given by

$$M_{SA}(\omega) = \exp\left\{-(2.1X|\omega|)^{5/3}(1 - V(Q,X)|\omega|^{1/3})\right\}. \tag{3}$$

If we denote by

- D: the system entrance pupil diameter (we recall from the geometrical optics that $D = f/N$ where f is the focal length and N the optics F-number),
- L: the path length (distance from the sensor to the observed scene),
- λ: the wavelength,
- C_n^2: the refractive index structure representing the turbulence intensity of the atmosphere [39],

we can define the following quantities: $k = \frac{2\pi}{\lambda}$, the coherence diameter r_0 defined by $r_0 = 2.1\rho_0$ where $\rho_0 = 1.437(k^2 L C_n^2)^{-3/5}$ (the coherence length) and $P = \sqrt{\lambda L}$. Then, the quantities Q, X and $V(Q,X)$ in eq. (3) can be defined by

$$Q = \frac{D}{P}, \quad X = \frac{D}{r_0}, \tag{4}$$

$$V(Q,X) = A + \frac{B}{10}\exp\left\{-\frac{(\gamma+1)^3}{3.5}\right\}, \tag{5}$$

where $\gamma = \log_{10}(X)$, $q = \log_2(Q)$ and

$$A = \begin{cases} 0.840 + 0.116\Sigma_{qa} & \text{with} \quad qa = 1.35(q+1.50) \\ & \text{if } q > -1.50 \\ 0.840 + 0.280\Sigma_{qc} & \text{with} \quad qc = 0.51(q+1.50) \\ & \text{if } q \leqslant -1.50 \end{cases}, \tag{6}$$

and $\Sigma_q = \frac{e^q - 1}{e^q + 1}$. The coefficient B is defined by

$$B = 0.805 + 0.265\Sigma_{qb} \quad \text{with} \quad qb = 1.45(q - 0.15). \tag{7}$$

Finally, in the Fourier domain, Fried's MTF $M_F(\omega)$ is defined by the product of $M_0(\omega)$ and $M_{SA}(\omega)$:

$$M_F(\omega) = M_0(\omega)M_{SA}(\omega). \tag{8}$$

Practically, this kernel depends on four parameters: D, L, λ and C_n^2. The first three clearly depend on the acquisition system and the imaging scene and are eventually known. The last parameter, C_n^2, represents the turbulence intensity and, accordingly to experimental measurements [39], is generally in the range $[10^{-16}m^{-2/3}, 10^{-12}m^{-2/3}]$ corresponding respectively to weak and strong turbulence. It requires some very specific complicated equipment to measure this parameter and unfortunately such equipment are impossible to use in real practical scenarios.

3 Atmospheric deconvolution

3.1 A simplified formulation of the Fried kernel

In this section, we propose a general kernel model of which the Fried kernel is a particular case. Our general formulation is given by

$$M_{\mathbf{a}}(\omega) = A(\omega)\exp\left\{\sum_{i=1}^{N} a_i f_i(\omega)\right\}, \tag{9}$$

where $N \in \mathbb{N}$, $A : \mathbb{R}^+ \to \mathbb{R}$ and $f_i : \mathbb{R}^+ \to \mathbb{R}$ are fixed and known while $\mathbf{a} = (a_1, \ldots, a_N)^T$ is a set of parameters to be estimated. We can notice that the Fried kernel corresponds to (the notation $a \asymp b$ mean "a plays the role of b")

$$\begin{gathered} A(\omega) = M_0(\omega), \quad N = 2, \\ a_1 \asymp (2.1X)^{5/3}, \quad a_2 \asymp (2.1X)^{5/3}V(Q,X), \\ f_1(.) = -(\cdot)^{5/3} \quad \text{and} \quad f_2(.) = (\cdot)^2. \end{gathered} \tag{10}$$

Note that this model also encompasses the traditional Gaussian kernel of bandwidth γ by choosing $A(\omega) = 1, N = 1, a_1 = \gamma^2/2$ and $f_1(\cdot) = -(\cdot)^2$.

3.2 A blind deconvolution model

In this paper, we model the spectrum y of a blurry image as

$$y(\omega) = M_{\mathbf{a}}(\omega) \cdot x(\omega) + \varepsilon(\omega), \tag{11}$$

where $M_{\mathbf{a}}$ is the unknown underlying Fried kernel, x the spectrum of the underlying clean image and ε is the spectrum of an additive white Gaussian noise component with standard deviation $\sigma > 0$. We consider the blind deconvolution problem of estimating jointly the image x and the vector of coefficients $\mathbf{a}$ solely from y. This blind deblurring problem is ill-posed such that regularization based on prior knowledge on the unknowns must be employed. To this end, we propose estimating x and $\mathbf{a}$ by looking for the image $x^\star$, the vector $\mathbf{a}^\star$ and an auxiliary variable $\lambda^* > 0$, minimizing the following regularized least square problem, defined for $\alpha > 0$, as

$$\underset{x,\mathbf{a},\lambda}{\operatorname{argmin}} \; \tfrac{1}{2\sigma^2} \|y - M_{\mathbf{a}} \cdot x\|_2^2 + R(x) + \lambda \|M_{\mathbf{a}}\|_2^2 + \frac{\alpha^2}{\lambda} \,. \tag{12}$$

The first term in (12) is the standard ℓ_2^2 data fidelity term that typically results from a Bayesian perspective when assuming that the noise component is additive white Gaussian with standard deviation σ. Though this term is convex with respect to $M_{\mathbf{a}}$ and x, it is not jointly convex with respect to both variables.

The second term $R(x) \geq 0$ is a regularization term enforcing $x^\star$ to fit some image prior knowledge. In this paper, we consider an image regularization term based on a zero-mean Gaussian Mixture Model (GMM) prior of natural image patches, as introduced in [40], and defined as

$$R(x) = -\sum_{i=1}^{n} \log \left[\sum_{i=1}^{K} w_k \mathcal{N}(\mathcal{P}_i \mathcal{F}^{-1} x; 0, \Sigma_k) \right], \tag{13}$$

where w_k and Σ_k are learned weights and covariance matrices, and $\mathcal{P}_i$ is the operator extracting in the spatial domain a discrete 8×8 zero-mean patch at pixel i. The Fourier transform and its inverse are denoted by $\mathcal{F}$ and $\mathcal{F}^{-1}$, respectively. Numerically, the Fourier transform is computed via the unitary discrete Fourier transform. This regularization term enforces all spatial patches of $x^\star$ (with overlap) to fit the GMM prior model. In practice, we use the GMM obtained in [40], composed of $K = 200$ components, trained by the Expectation-Maximization algorithm on a collection of 2 million patches extracted from the Berkeley Segmentation Data Set (BSDS) [41]. Due to the multi-modality of GMMs, this regularization term is highly non-convex.

The third and fourth terms in (12) act jointly on the structure of $M_{\mathbf{a}^\star}$. Since $R(x)$ always get smaller as the images become smoother, it subsequently favors the Fried kernel that over-amplifies high frequencies. To counter-balance this effect, we prevent the kernel from taking arbitrarily large values by controlling for its total energy $\|M_{\mathbf{a}^\star}\|_2^2$. The variable $\lambda > 0$ plays the role of a regularization parameter: the larger λ, the smaller $\|M_{\mathbf{a}^\star}\|_2^2$ is expected to be. We observed that this parameter cannot be blindly chosen as a fixed constant for all turbulence scenarios. For this reason, we let this variable be optimized in a way that it cannot get too small as controlled by the last term in (12) and parameterized by $\alpha > 0$. Though convex in λ and $M_{\mathbf{a}}$, this problem is neither convex in $\mathbf{a}$, nor jointly in λ and $\mathbf{a}$.

3.3 A blind deconvolution optimization procedure

Due to the non-convexity of the joint optimization problem, we adopt in this paper a greedy approach that we coin Blind Atmospheric TUrbulence Deconvolution (BATUD). BATUD attempts to minimize (12) by using the following alternate minimization scheme

$$x^{t+1} \in \underset{x}{\operatorname{argmin}} \tfrac{1}{2\sigma^2} \|y - M_{\mathbf{a}^t} \cdot x\|_2^2 + R(x), \tag{14}$$

$$\lambda^{t+1} \in \underset{\lambda}{\operatorname{argmin}} \; \lambda \|M_{\mathbf{a}^t}\|_2^2 + \frac{\alpha^2}{\lambda}, \tag{15}$$

$$\mathbf{a}^{t+1} \in \underset{\mathbf{a}}{\operatorname{argmin}} \tfrac{1}{2\sigma^2} \left\| y - M_{\mathbf{a}} \cdot x^{t+1} \right\|_2^2 + \lambda^{t+1} \|M_{\mathbf{a}}\|_2^2, \tag{16}$$

where $t \geq 0$ is the time step and the initialization for $\mathbf{a}^0$ will be discussed in Section 4.1 (note that we do not need to initialize x^0 and λ^0).

Problem (14) can be seen as a non-blind deconvolution problem for the fixed convolutional kernel $M_{\mathbf{a}^t}$. This deconvolution problem, with $R(\cdot)$ the GMM patch based prior given in (13), can be solved with the Expected Patch Log-Likelihood (EPLL) algorithm [40]. EPLL is an iterative algorithm that repeats three steps: (i) extracting all image patches, (ii) filtering all patches assuming a GMM patch prior, and (iii) performing a Wiener-like deconvolution step that combines all previously estimated patches. In this paper, we use Fast EPLL (FEPLL), a fast approximation of EPLL introduced in [42]. The corresponding code, provided by the authors, is freely available on GitHub*.

Problem (15) is convex and has a trivial closed form solution given by

$$\lambda^{t+1} = \frac{\alpha}{\|M_{\mathbf{a}^t}\|_2} \,. \tag{17}$$

Equation (17) shows that if $\|M_{\mathbf{a}^t}\|_2$ gets larger with t, λ^t decreases until a trade-off is achieved.

Problem (16) can be recast as the minimization of

$$E(\mathbf{a}) = \int \tfrac{1}{2} |M_{\mathbf{a}}(\omega) x(\omega)|^2 - x(\omega)^* M_{\mathbf{a}}(\omega)^* y(\omega) + \lambda \sigma^2 |M_{\mathbf{a}}(\omega)|^2 \, \mathrm{d}\omega, \tag{18}$$

where * denotes the complex conjugate, and we removed the dependency of x and λ with t for the sake of notation simplicity. Equation (18) can be minimized by Newton's descent consisting in iterating the following update

$$\begin{pmatrix} a_1 \\ \vdots \\ a_N \end{pmatrix} \leftarrow \begin{pmatrix} a_1 \\ \vdots \\ a_N \end{pmatrix} - \begin{pmatrix} \frac{\partial^2 E(\mathbf{a})}{\partial a_1 \partial a_1} & \cdots & \frac{\partial^2 E(\mathbf{a})}{\partial a_1 \partial a_N} \\ \vdots & & \vdots \\ \frac{\partial^2 E(\mathbf{a})}{\partial a_N \partial a_1} & \cdots & \frac{\partial^2 E(\mathbf{a})}{\partial a_N \partial a_N} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial E(\mathbf{a})}{\partial a_1} \\ \vdots \\ \frac{\partial E(\mathbf{a})}{\partial a_N} \end{pmatrix}, \tag{19}$$

where

$$\frac{\partial E(\mathbf{a})}{\partial a_i} = \int f_i(\omega) x(\omega)^* M_{\mathbf{a}}(\omega)^* [M_{\mathbf{a}}(\omega) x(\omega) - y(\omega)] + \lambda \sigma^2 f_i(\omega) |M_{\mathbf{a}}(\omega)|^2 \, \mathrm{d}\omega, \tag{20}$$

and

$$\frac{\partial^2 E(\mathbf{a})}{\partial a_i \partial a_j} = \int f_i(\omega) f_j(\omega) x(\omega)^* M_{\mathbf{a}}(\omega)^* [2 M_{\mathbf{a}}(\omega) x(\omega) - y(\omega)] + 2\lambda \sigma^2 f_i(\omega) f_j(\omega) |M_{\mathbf{a}}(\omega)|^2 \, \mathrm{d}\omega \,. \tag{21}$$

In practice, at every time step $t > 1$, we use a warm start strategy to initialize Newton's descent by using the solution $\mathbf{a}^{t-1}$ obtained at the previous time step. By doing so, we observed a very fast

*https://goo.gl/xjqKUA

convergence of this numerical scheme. Though the objective function might have regions of non-convexity where its Hessian matrix is not non-negative definite, we always observed a stable behavior of this algorithm by using such an initialization. Nevertheless, as a safeguard, if any numerical issues were arising from the resolution of (19), the optimization of $\mathbf{a}$ would be instead carried out by the simplex search method of Lagarias *et al.* [43][†].

3.4 Stopping criteria

Regarding the overall optimization scheme, at each time step $t \geq 1$ we compute the relative errors between two consecutive estimates for the three quantities to be optimized as

$$\delta_x^t = \frac{\|x^t - x^{t-1}\|_2}{\|x^{t-1}\|_2},\ \delta_\lambda^t = \frac{|\lambda^t - \lambda^{t-1}|}{|\lambda^{t-1}|}, \tag{22}$$

$$\text{and} \quad \delta_{\mathbf{a}}^t = \frac{\|M_{\mathbf{a}^t} - M_{\mathbf{a}^{t-1}}\|_2}{\|M_{\mathbf{a}^{t-1}}\|_2}. \tag{23}$$

We consider that the algorithm has converged as soon as these relative errors are under a threshold τ during at least Δ consecutive iterations. In other words, we stop the algorithm at time step t when

$$\max_{t-\Delta \leq t' \leq t} \max(\delta_x^{t'}, \delta_\lambda^{t'}, \delta_{\mathbf{a}}^{t'}) < \tau. \tag{24}$$

Similarly, for Problem (16), we run Newton's descent by iteratively applying eq. (19) until two consecutive estimates for $\mathbf{a}$ achieve a relative error smaller than τ.

4 Experiments

In this section, we conducted several studies and experiments to understand the behavior and demonstrate the performance of our proposed blind atmospheric turbulence deconvolution algorithm. For the sake of reproducible research, our Matlab implementation of BATUD has been made open-source and is freely available online at:

`https://www.charles-deledalle.fr/batud`

4.1 Parameter setting and initialization

In practice, we choose $\alpha^2 = \gamma n / \sigma^2$ with a constant parameter $\gamma = 8.5 \cdot 10^{-3}$ that is kept fixed in all our experiments, and n being the number of pixels. The initial blur parameter $\mathbf{a}^0$ has been chosen empirically to correspond to the Fried kernel with $X = 2.75$ and $V = 1.06$. FEPLL [42] was used with default parameters which consists in performing five iterations of patch filtering and aggregation. On experiments involving real data, we estimate the noise standard deviation σ using a classical procedure based on the Median Absolute Deviation (MAD) of detail coefficients, as explained in [44]. Otherwise specified, all experiments have been carried out by performing at most 150 iterations of the proposed alternate optimization scheme with the stopping criterion parameterized by $\tau = 2 \cdot 10^{-4}$ and $\Delta = 3$.

4.2 Experimental study on the simulated scenarios

We start to assess the proposed approach on simulated blurs applied to the Kodak dataset[*] composed of 23 images of size 768×512 (or 512×768) and encoded on 8-bits and converted to grayscale. We apply three types of blur, by using the original Fried kernel model given in Section 2, corresponding to different levels of turbulence: Weak ($X = 2.25$, $V = 1.29$), Medium ($X = 3.00$, $V = 1.00$) and Strong ($X = 5.00$, $V = 0.80$). In addition we add white Gaussian noise with three different levels: $\sigma_1 = 1.3$, $\sigma_2 = 1.9$ and $\sigma_3 = 2.5$. Note that this choice of levels of turbulence and noise captures the range of scenarios we observed in practice. Our deconvolution procedure is then applied to each simulated image. Quantitative performance measurements are computed by calculating the Peak Signal to Noise Ratio (PSNR) criterion defined in decibels (dB) by

$$\mathrm{PSNR}(\hat{x}, x) = 10 \log_{10} \frac{255^2}{\frac{1}{n}\|\hat{x} - x\|_2^2}, \tag{25}$$

where $\hat{x}$ denotes the restored image and x the reference image (both images are assumed to be encoded on 8-bits). We also consider Structural Similarity Index (SSIM) introduced in [45] that provides a quantitative measure of quality between 0 and 1, and supposedly being more faithful to visual perception than the PSNR is. The larger these deblurring criteria are, the better the deblurring quality is considered to be.

We also assess the performance of the algorithm to estimate the original kernel used in the simulations. To quantitatively measure this performance, we compute the relative error between the original and estimated kernels (we will call it the Kernel Relative Error, i.e KRE) defined by

$$\mathrm{KRE}(\hat{\mathbf{a}}, \mathbf{a}) = \frac{\|M_{\hat{\mathbf{a}}} - M_{\mathbf{a}}\|_2^2}{\|M_{\mathbf{a}}\|_2^2}, \tag{26}$$

Table 1 Simulated atmospheric turbulence. PSNR, SSIM and Kernel relative errors (KRE) averaged on the Kodak dataset

		Weak			Medium			Strong		
		σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
PSNR	Blurry	30.45	30.28	30.05	26.75	26.67	26.57	24.39	24.35	24.29
	BATUD	34.53	34.63	34.12	32.37	31.27	30.52	27.44	27.13	26.88
	Oracle	*41.21*	*38.97*	*37.36*	*33.29*	*31.99*	*31.15*	*27.71*	*27.38*	*27.14*
SSIM	Blurry	0.858	0.834	0.810	0.705	0.686	0.667	0.482	0.469	0.454
	BATUD	0.925	0.902	0.879	0.863	0.822	0.789	0.694	0.658	0.629
	Oracle	*0.932*	*0.905*	*0.882*	*0.868*	*0.831*	*0.800*	*0.709*	*0.674*	*0.646*
KRE	BATUD	0.165	0.162	0.164	0.023	0.032	0.040	0.020	0.024	0.028
	Oracle	*0.064*	*0.107*	*0.147*	*0.003*	*0.005*	*0.008*	*0.000*	*0.000*	*0.001*

4.2.1 Behavioral study of the proposed algorithm:

In order to assess the quality of BATUD (our blind deconvolution approach), we computed the oracle images, i.e the deconvolved images using the real kernel which was used in the simulation (this is basically equivalent to perform a non-blind deconvolution). In addition, we also computed oracle kernels, i.e the estimated kernels using the original clean image used in the simulation. The performance of these oracle estimators provides in some sense an ideal baseline for the expected performance of our blind deconvolution algorithm.

Comparisons in terms of PSNR, SSIM and KRE (averaged over the 23 images) are given in Table 1 for the three levels of turbulence and the three levels of noise we considered. Visual comparisons on two images of the dataset are provided in Figs. 1 and 2 for the three considered levels of turbulence, and the two levels of noise σ_1 and σ_3, respectively. Additional results obtained from the BATUD algorithm on different simulated scenarios are also provided in Fig. 3.

For Medium and Strong turbulence levels, we can see that the estimated kernels have, in average, relative errors of no more than 4%. Given that the Fried kernels are well estimated in this case, the deblurring results are close to the one provided by the non-blind oracle, with a drop of PSNR smaller than 1dB and a drop of SSIM smaller than 0.03. Visual inspections of the results of blind and non-blind deconvolution, in Medium and Strong turbulence levels (second and third column in the figures), show that these drops of quality measures are indeed visually imperceptible.

In the Weak turbulence level, the estimation of the Fried kernel is less accurate than in the other two scenarios. The relative errors

[†] *using the Matlab function* `fminsearch`.
[*] `http://r0k.us/graphics/kodak/`

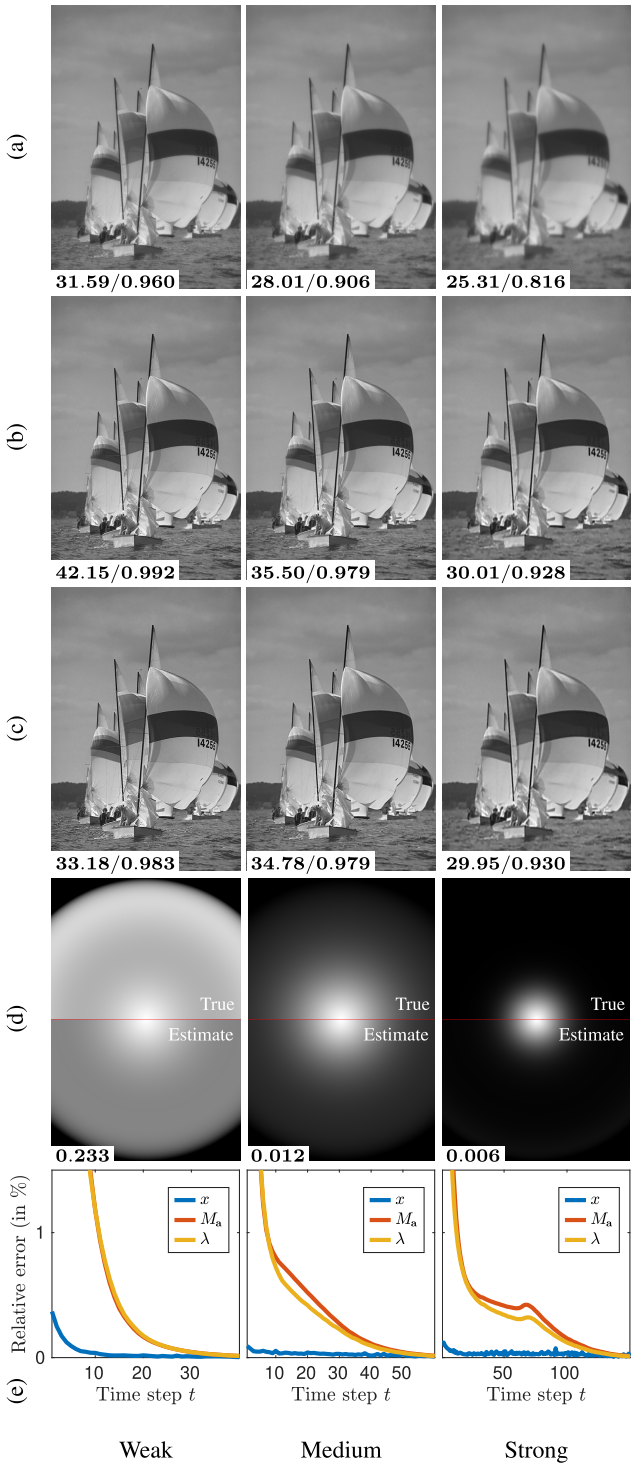

Fig. 1: (a) Blurry image y under medium noise level $\sigma = 0.005$. (b) Oracle deconvolved image with BATUD. (c) Deconvolved image with BATUD. (d) True and estimated Fried kernel with the proposed approach (the zero frequency is at the center and amplitudes are displayed in log-scale). (e) Relative errors between consecutive estimates with respect to the time step t. (a-c) PSNR and SSIM values are displayed on the bottom-left. (d) KRE is displayed on the bottom-left.

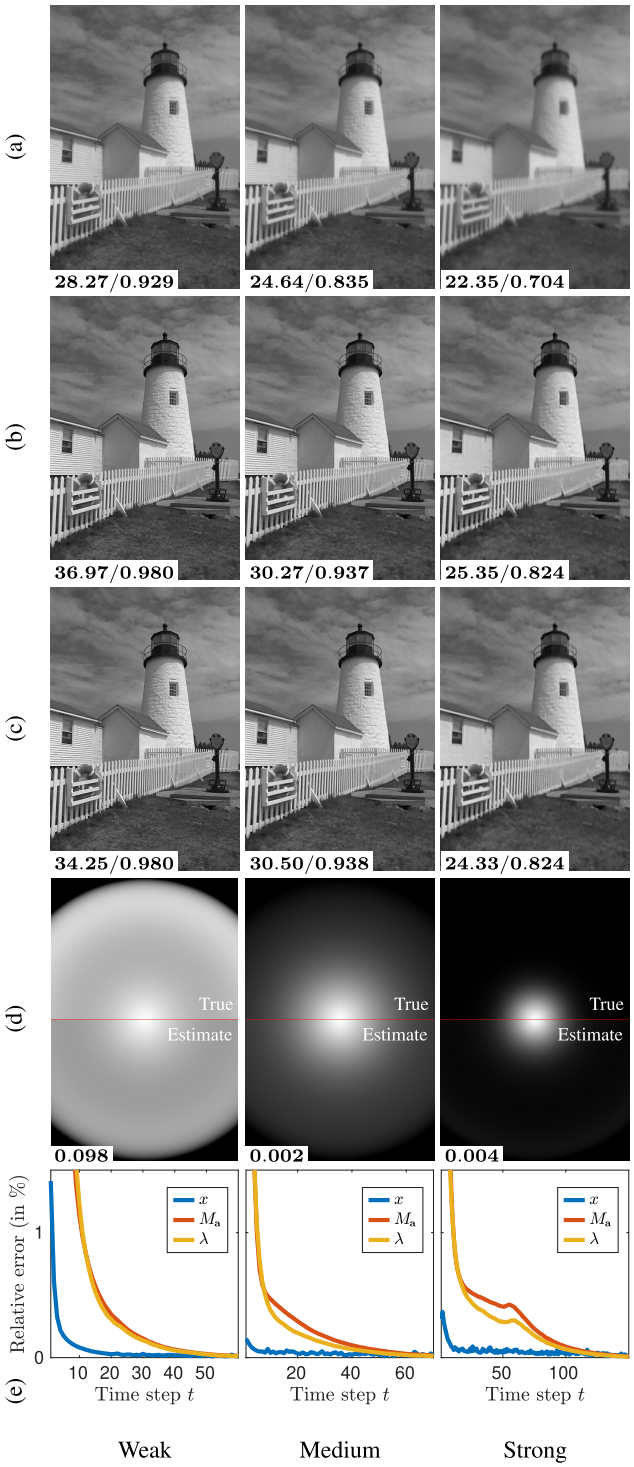

Fig. 2: (a) Blurry image y under strong noise level $\sigma = 0.01$. (b) Oracle deconvolved image with FEPLL. (c) Deconvolved image with BATUD. (d) True and estimated Fried kernel with the proposed approach (the zero frequency is at the center and amplitudes are displayed in log-scale). (e) Relative errors between consecutive estimates with respect to the time step t. (a-c) PSNR and SSIM values are displayed on the bottom-left. (d) KRE is displayed on the bottom-left.

of BATUD increase around 16%. Even the oracle kernel estimator, that was providing relative errors smaller than 1% in the other two scenarios, now provides relative errors around 10%. Inspecting our estimation of the Fried kernel in Fig. 1 (first column) shows that our approach tends to slightly under-estimate the high-frequency responses of the filter (leading to a KRE of about 23% in this case). As a result the deblurring procedure over-amplifies these high frequencies leading to structures appearing sharper than they are supposed to be. Small Gibbs-like artifacts can also be observed around high-contrasted objects. These effects can be seen in Fig. 1 (first column) by comparing the ability of the BATUD and the non-blind deconvolution at recovering the waves patterns, the folds and the

numbers on the sail. Though barely visible, Table 1 shows that these over-sharpening effects and Gibbs artifacts contribute to a loss of PSNR of about 7dB compared to the non-blind oracle. Nevertheless, compared to the oracle, these artifacts do not influence much the SSIM, the drop being smaller than 0.01, hence reflecting our visual inspections.

As an additional study, Figs. 1 and 2 (last rows) provide the evolution of the relative errors (in percentage) between two consecutive estimates with respect to the iteration index t of BATUD. These quantities correspond to δ_x^t, δ_λ^t and $\delta_{M_\mathbf{a}}^t$ defined in (22). The curves are computed until the algorithm stops when the stopping criterion in (24) is satisfied. We first observe that under any turbulence and noise scenario, these relative errors converges to the required level of tolerance 0.02% in at most 150 iterations. Under Weak and Medium levels of turbulence, the procedure takes at most 70 iterations and the relative errors for λ and $M_\mathbf{a}$ appears to be monotonically non-increasing. Under a Strong level of turbulence, BATUD behaves differently. It converges in about 150 iterations and presents two different phases. The first $\sim$65 iterations can be seen as an exploration phase where λ and $M_\mathbf{a}$ keep changing (two consecutive estimates differ more than 25%) until they reach a probing configuration. Afterwards, the procedure enters an exploitation phase in which this promising configuration is iteratively refined until no significant changes arise.

Given that one iteration of BATUD takes about 1.4s to process a Kodak image*, the computation time ranges from 56s to 3min 30s per image depending on the underlying level of turbulence.

4.2.2 Comparisons with state-of-the-art blind deconvolution algorithms: In this section, we compare our algorithm against several standard and state-of-the-art algorithms. The first algorithm, directly available in Matlab via the *deconvblind* function, is based on a Maximum Likelihood estimator to recover both the convolution kernel and the restored image [46, 47] (we will denote it MLE). The second algorithm (denoted OBD, available at†) described in [48] (this algorithm was initially designed to process a set of images but the authors mentioned that it also works using a single image), minimizes a loss function based on ℓ_2 assuming a Poisson model on the data and the positivity of both the data and the kernel. In [49], an estimation method (denoted KOT, whose code is available at‡) based on a modified version of the Automatic Relevance Determination model is proposed. The authors include a Student's $t-$prior as well as some Gamma distribution within a variational Bayesian framework to solve this model. Cai et al., in [50], propose a blind deconvolution algorithm using a variational approach (we will denote this algorithm CLS and use the code available at§). The authors use a sparsity assumption in a framelet domain for both the reconstructed image and the kernel, an ℓ_2 constraint is also added to the kernel. In [51], Shan et al. propose a unified probabilistic model to estimate both the kernel and the latent image (this algorithm will be denoted SJA and is available at¶). They use a Gaussian model for the noise, an exponential prior for the kernel and a combination of a local and global priors for the image. This algorithm has been used in actual turbulence mitigation techniques in [14, 20] and [27]. Finally, the last algorithm (denoted ZWZ and available at‖) we consider combines probabilistic and variational models (an ℓ_2 and ℓ_p, with $0 < p < 1$, constraints are used to enforce a sparsity prior). A Maximum A Posteriori approach is then used to numerically solve this model. For all these algorithms, we used the default parameters recommended by the respective authors.

Table 2 Comparison with state-of-the-art blind deconvolution algorithms. PSNR and SSIM averaged on the Kodak dataset.

		Weak			Medium			Strong		
		σ_1	σ_2	σ_3	σ_1	σ_2	σ_3	σ_1	σ_2	σ_3
PSNR	Blurry	30.45	30.28	30.05	26.75	26.67	26.57	24.39	24.35	24.29
	BATUD	**34.53**	**34.63**	**34.12**	**32.37**	**31.27**	**30.52**	**27.44**	**27.13**	**26.88**
	CLS	26.26	26.15	25.82	30.14	29.12	28.29	26.62	26.43	26.14
	KOT	27.70	27.57	27.57	28.18	28.02	27.97	25.89	25.79	25.68
	MLE	30.45	30.28	30.05	26.75	26.67	26.57	24.39	24.35	24.29
	OBD	30.45	30.28	30.05	26.75	26.67	26.57	24.39	24.35	24.29
	SJA	26.52	25.49	23.86	26.47	25.93	24.08	24.89	24.09	22.28
	ZWZ	31.26	31.03	30.19	28.51	28.13	27.78	25.83	25.76	25.63
SSIM	Blurry	0.943	0.940	0.935	0.863	0.859	0.854	0.734	0.731	0.726
	BATUD	**0.987**	**0.984**	**0.978**	**0.966**	**0.949**	**0.934**	**0.869**	**0.853**	**0.839**
	CLS	0.864	0.857	0.847	0.949	0.931	0.912	0.851	0.839	0.824
	KOT	0.923	0.915	0.907	0.913	0.904	0.896	0.820	0.809	0.798
	MLE	0.943	0.940	0.935	0.863	0.859	0.854	0.734	0.731	0.726
	OBD	0.943	0.940	0.935	0.863	0.859	0.854	0.734	0.731	0.726
	SJA	0.909	0.877	0.840	0.844	0.818	0.724	0.780	0.718	0.562
	ZWZ	0.910	0.904	0.894	0.914	0.900	0.887	0.807	0.800	0.790

We run these algorithms on the Kodak dataset used in Section 4.1 and re-utilize the PSNR and SSIM metrics (unfortunately some of the available codes do not return the estimated kernels thus we were not able to use the KRE metric). Table 2 provides the averaged PSNR and SSIM for each algorithm. It is clear that our method outperforms the state-of-the-art algorithms.

We observe that CLS and ZWZ locally modify the image dynamics and introduce some ringing artifacts (especially in the case of weak blur). Moreover, CLS also performs poorly (i.e limited deblurring) on the weak turbulence case, this is probably due to the fact the algorithm is made to remove motion blur hence it is not capable of estimating small isotropic type kernels. The SJA algorithm is very sensitive to the presence of noise that leads to the introduction of a lot of artifacts in such case. Kotera is not capable to deal with strong blurs and sometimes it tends to do some image sharpening than actual deblurring. Some of the observed artifacts for the CLS, KOT, SJA and ZWZ algorithms are illustrated in Fig.4. The MLE and OBD algorithms are inefficient at estimating the underlying kernel and therefore almost do not perform any deblurring at all. If these algorithms perform reasonably well on some images, in average, they perform worse than keeping the actual blurry images! These results clearly show that a deconvolution model based on a physical model is preferable to a blind algorithm.

Moreover, note that the CLS and ZWZ algorithms take about 8min and 5min to process a Kodak image*, respectively, while BATUD takes only about 2min.

4.3 Experiments on real images

In this section, we experiment the BATUD algorithm on real images coming from several datasets: the OTIS dataset [52], the NATO-SET152 New Mexico field trials, the U.S. Army RDECOM CERDEC Night Vision & Electronic Sensors Directorate data and the test images available from the IPOL demonstration webpage†. The sequences from these datasets are first stabilized using the Mao-Gilles algorithm [28, 29] to obtain the input image y. These images and their deconvolved versions, as well as the estimated kernel are provided in Fig. 5. We can observe that the algorithm is capable of removing the blur and retrieve more details in the images (the texts are easier to read, most textures are recovered and thin structures

Computation time evaluated on an Intel(R) Xeon(R) CPU E5-2640 v2 @ 2.00GHz, using one single core only.

†`http://pixel.kyb.tuebingen.mpg.de/obd/`

‡`http://staff.utia.cas.cz/kotera/projects/model_discrepencies_17/`

§`https://blog.nus.edu.sg/matjh/files/2019/01/BlindDeblurSingleTIP-2jrncsd.zip`

¶`http://www.cse.cuhk.edu.hk/~leojia/projects/motion_deblurring/`

‖`https://drive.google.com/file/d/0BzoBvkfRHe5bUF9jQ1ZsWXRYSkk/edit?usp=sharing`

We provide only approximate values since the CLS and ZWZ implementations perform the reading and writing of images within their code.

†`http://demo.ipol.im/demo/46/`

Fig. 3: BATUD results on different simulated scenarios of blur and noise from the Kodak dataset. PSNR and SSIM values are displayed on the bottom-left.

are sharpened). As expected, we can also notice that the stronger the turbulence, the narrower are the estimated kernels.

In Fig. 6, we compare our algorithm against the CLS and ZWZ blind deconvolution algorithms mentioned in Section 4.2.2 (we choose only these two algorithms since they are the ones giving the

Fig. 4: Illustration of artifacts appearing with the CLS, KOT, SJA and ZWZ blind deblurring algorithms.

best results in medium or strong turbulence). On the first image, we can observe that the CLS algorithm provides a slightly better result (especially in the textured regions) than ZWZ and BATUD while for the two other images BATUD gives superior results. In particular, on the second image, the word "*A BEAR*" is easier to read from the BATUD output. The same observation can be made on the third image where BATUD is capable of better restoring smaller text than the other algorithms.

5 Conclusion

We introduced BATUD a new algorithm for blind atmospheric turbulence deconvolution. Unlike concurrent algorithms, ours relies on a physical model of blur induced by atmospheric turbulence, namely the Fried kernel. Our approach is fast and fully automatic as it estimates jointly the deconvolved image and the Fried kernel while auto-adjusting the regularization parameter. Experiments on simulated and real data have demonstrated the improved performance of BATUD in several atmospheric turbulence scenarios against state-of-the-art approaches. In future works, we plan to develop algorithmic recipes to make our implementation even faster and to port the implementation on Graphics Processing Units (GPUs). We also aim at adapting the approach to low-light vision systems and hyperspectral imagery. Extensions of this work to submarine imagery by using appropriate physical models of underwater turbulence are also under investigation.

6 Acknowledgments

This work was partially supported by the Air Force Office of Scientific Research under the grant number FA9550-15-1-0065. The authors want to thank the members of the NATO SET156 (ex-SET072) Task Group and the U.S. Army RDECOM CERDEC Night Vision & Electronic Sensors Directorate for the opportunity of using their respective data. We also want to thanks the different authors who made their code publicly available.

7 References

1 Frakes, D.H., Monaco, J.W., Smith, M.J.T.: 'Suppression of atmospheric turbulence in video using an adaptive control grid interpolation approach', *IEEE Conference Proceedings on Acoustics, Speech and Signal Processing*, 2001, **3**, pp. 1881–1884

2 Patel, A., Israni, D., Kumar, N.A.M., Bhatt, C.: 'An adaptive image registration technique to remove atmospheric turbulence', *Statistics, Optimization And Information Computing*, 2019, **7**, pp. 439–446

3 Gepshtein, S., Shtainman, A., Fishbain, B. 'Restoration of atmospheric turbulent video containing real motion using rank filtering and elastic image registration'. In: Proceedings of EUSIPCO. (Vienna, Austria, 2004.

4 Lemaitre, M. 'Etude de la turbulence atmosphérique en vision horizontale lointaine et restauration de séquences dégradées dans le visible et l'infrarouge'. Université de Bourgogne, 2007

5 Micheli, M., Lou, Y., Soatto, S., Bertozzi, A.: 'A linear systems approach to imaging through turbulence', *Journal of Mathematical Imaging and Vision*, 2014, **48**, (1), pp. 185–201

6 Gilboa, G., Osher, S.: 'Nonlocal operators with applications to image processing', *Multiscale Model Sim*, 2008, **7**, pp. 1005–1028

7 Micheli, M. 'The centroid method for imaging through turbulence'. (MAP5, Université Paris Descartes, 2012.

8 Meinhardt.Llopis, E., Micheli, M.: 'Implementation of the Centroid Method for the Correction of Turbulence', *Image Processing On Line*, 2014, **4**, pp. 187–195

9 Li, D., Mersereau, R.M., Simske, S.: 'Atmospheric turbulence-degraded image restoration using principal components analysis', *IEEE Geoscience and Remote Sensing Letters*, 2007, **4**, pp. 340–344

10 Song, C., Ma, K., Li, A., Chen, X., Xu, X.: 'Diffraction-limited image reconstruction with SURE for atmospheric turbulence removal', *Infrared Physics & Technology*, 2015, **71**, pp. 171–174

11 Aubailly, M., Vorontsov, M.A., Carhat, G.W., Valley, M.T. 'Automated video enhancement from a stream of atmospherically-distorted images: the lucky-region fusion approach'. In: Proceedings of SPIE. vol. 7463. (, 2009.

12 Yang, B., Zhang, W., Xie, Y., Li, Q. 'Distorted image restoration via non-rigid registration and lucky-region fusion approach'. In: Proceedings of the Third International Conference on Information Science and Technology. (Yangzhou, Jiangsu, China, 2013. pp. 414–418

Fig. 5: Results obtained on height images with real atmospheric turbulence. (a) Blurry images y with different levels of turbulence. (b) Deconvolved images. (c) Estimated Fried kernels (the zero frequency is at the center and amplitudes are displayed in log-scale).

13 Gilles, J., Dagobert, T., Franchis, C.D. 'Atmospheric turbulence restoration by diffeomorphic image registration and blind deconvolution'. In: Advanced Concepts for Intelligent Vision Systems - 10th International Conference, ACIVS, Proceedings. vol. 5259 of *Lecture Notes in Computer Science*. (Juan-les-Pins, France: Springer, 2008. pp. 400–409

14 Zhu, X., Milanfar, P. 'Image reconstruction from videos distorted by atmospheric turbulence'. In: SPIE Electronic Imaging, Conference 7543 on Visual Information Processing and Communication. (San Jose, CA, USA, 2010.

15 Halder, K.K., Tahtali, M., Anavatti, S.G. 'A fast restoration method for atmospheric turbulence degraded images using non-rigid image registration'. In: 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI). (Mysore, 2013. pp. 394–399

16 Halder, K.K., Tahtali, M., Anavatti, S.G. 'High precision restoration method for non-uniformly warped images'. In: Advanced Concepts for Intelligent Vision Systems - 15th International Conference, ACIVS, Proceedings. vol. 8192 of *Lecture Notes in Computer Science*. (Poznan, Poland: Springer, 2013. pp. 60–67

Fig. 6: Comparison of the proposed approach against the CLS and ZWZ blind-deconvolution algorithms on real images.

17 Halder, K.K., Tahtali, M., Anavatti, S.G.: 'Model-free prediction of atmospheric warp based on artificial neural network', *Applied Optics*, 2014, **53**, (30), pp. 7087–7094

18 Bai, X., Liu, M., He, C., Dong, L., Zhao, Y., Liu, X. 'Restoration of turbulence-degraded images based on deep convolutional network'. In: Zelinski, M.E., Taha, T.M., Howe, J., Awwal, A.A.S., Iftekharuddin, K.M., editors. Proceedings of SPIE Optical Engineering + Applications. vol. 11139 of *Applications of Machine Learning*. (San Diego, USA, 2019. pp. 111390B–1–111390B–9

19 Anantrasirichai, N., Achim, A., Kingsbury, N.G., Bull, D.R.: 'Atmospheric turbulence mitigation using complex wavelet-based fusion', *IEEE Transaction in Image Processing*, 2013, **22**, (6), pp. 2398–2407

20 Xie, Y., Zhang, W., Tao, D., Hu, W., Qu, Y., Wang, H.: 'Removing turbulence effect via hybrid total variation and deformation-guided kernel regression', *IEEE Transaction on Image Processing*, 2016, **25**, (10), pp. 4943–4958

21 Lau, C.P., Lai, Y.H., Lui, L.M.: 'Variational models for joint subsampling and reconstruction of turbulence-degraded images', *Journal of Scientific Computing*, 2019, **78**, (3), pp. 1488–1525

22 Nichols, J.M., Emerson, T.H., Cattell, L., Park, S., Kanaev, A., Bucholtz, F., et al.: 'Transport-based model for turbulence-corrupted imagery', *Applied Optics*, 2018, **57**, (16), pp. 4524–4536

23 Gal, R., Kiryati, N., Sochen, N.: 'Progress in the restoration of image sequences degraded by atmospheric turbulence', *Pattern Recognition Letters*, 2014, **48**, pp. 8–14

24 Nieuwenhuizen, R., Schutte, K. 'Deep learning for software-based turbulence mitigation in long-range imaging'. In: Dijk, J., editor. Artificial Intelligence and Machine Learning in Defense Applications. vol. 11169. International Society for Optics and Photonics. (SPIE, 2019. pp. 153–162

25 Arora, R., Singh, G.: 'Key challenges and resolutions for atmospheric turbulence effect removal', *International Journal of Grid and Distributed Computing*, 2018, **11**, (11), pp. 39–52

26 Patel, K., Israni, D., Garg, D.: 'A roadmap to mitigation techniques: Bedrock for atmospheric turbulence', *International Journal on Advanced Networking and Applications*, 2019, **10**, (5), pp. 4007–4012

27 Lau, C.P., Lai, Y.H., Lui, L.M.: 'Restoration of atmospheric turbulence-distorted images via RPCA and quasiconformal maps', *Inverse Problems*, 2019, **35**, (7), pp. 074002

28 Mao, Y., Gilles, J.: 'Non rigid geometric distortions correction - application to atmospheric turbulence stabilization', *Journal of Inverse Problems and Imaging*, 2012, **6**, (3), pp. 531–546

29 Gilles, J.: 'Mao-Gilles Stabilization Algorithm', *Image Processing On Line*, 2013, **2013**, pp. 173–182

30 Zwart, C.M., Pracht, R.J., Frakes, D.H. 'Improved motion estimation for restoring turbulence distorted video'. In: Proceedings of SPIE. vol. 8355 of *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXIII*. (, 2012. pp. 83550E–1–83550E–9

31 Gezer, G., Mercimek, M. 'An investigation of registration methods on the images degraded by atmospheric turbulence'. In: 2018 6th International Conference on Control Engineering and Information Technology (CEIT). (Istanbul, Turkey: IEEE, 2018.

32 Hirsch, M., Sra, S., Scholkopf, B., Harmeling, S. 'Efficient filter flow for space-variant multiframe blind deconvolution'. In: Computer Vision and Pattern Recognition Conference. (, 2010.

33 Li, D., Simske, S.: 'Atmospheric turbulence degraded image restoration by kurtosis minimization', *IEEE Geoscience and Remote Sensing Letters*, 2009, **6**, (2), pp. 244–247

34 Lou, Y., Kang, S.H., Soatto, S., Bertozzi, A.L.: 'Video stabilization of atmospheric turbulence distortion', *Inverse Problems and Imaging*, 2013, **7**, (3), pp. 839–861

35 Gilles, J., Osher, S.: 'Wavelet burst accumulation for turbulence mitigation', *Journal of Electronic Imaging*, 2016, **25**, (3), pp. 033003–1–033003–9

36 Fried, D.L.: 'Optical resolution through a randomly inhomogeneous medium for very long and very short exposures', *Journal of The Optical Society of America*, 1966, **56**, (10), pp. 1372–1379

37 Tofsted, D.H.: 'Reanalysis of turbulence effects on short-exposure passive imaging', *Optical Engineering*, 2011, **50**, (1), pp. 1–9

38 Gilles, J., Osher, S. 'Fried deconvolution'. In: SPIE Defense, Security and Sensing conference. (Baltimore, US, 2012.

39 Tunick, A., Tikhonov, N., Vorontsov, M., Carhart, G. 'Characterization of optical turbulence (C_n^2) data measured at the ARL A_LOT facility'. (Army Research Lab, 2005. ARL-MR-625

40 Zoran, D., Weiss, Y. 'From learning models of natural image patches to whole image restoration'. In: 2011 International Conference on Computer Vision. (IEEE, 2011. pp. 479–486

41 Martin, D., Fowlkes, C., Tal, D., Malik, J. 'A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics'. In: Proc. 8th Int'l Conf. Computer Vision. vol. 2. (, 2001. pp. 416–423

42 Parameswaran, S., Deledalle, C.A., Denis, L., Nguyen, T.Q.: 'Accelerating GMM-Based Patch Priors for Image Restoration: Three Ingredients for a 100× Speed-Up', *IEEE Transaction on Image Processing*, 2019, **28**, (2), pp. 687–698

43 Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E.: 'Convergence properties of the nelder–mead simplex method in low dimensions', *SIAM Journal on optimization*, 1998, **9**, (1), pp. 112–147

44 Donoho, D.L., Johnstone, J.M.: 'Ideal spatial adaptation by wavelet shrinkage', *biometrika*, 1994, **81**, (3), pp. 425–455

45 Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., et al.: 'Image quality assessment: from error visibility to structural similarity', *IEEE transactions on image processing*, 2004, **13**, (4), pp. 600–612

46 Holmes, T.J., Bhattacharyya, S., Cooper, J.A., Hanzel, D., Krishnamurthi, V., Lin, W.c., et al. In: Pawley, J.B., editor. 'Light microscopic images reconstructed by maximum likelihood deconvolution'. (Boston, MA: Springer US, 1995. pp. 389–402

47 Hanisch, R.J., White, R.L., Gilliland, R.L. In: Jansson, P.A., editor. 'Deconvolutions of Hubble space telescope images and spectra'. (CA: Academic Press, 1997. pp. 310–360

48 Harmeling, S., Hirsch, M., Sra, S., Schölkopf, B. 'Online blind deconvolution for astronomical imaging'. In: 2009 IEEE International Conference on Computational Photography (ICCP). (San Francisco, CA, USA: IEEE, 2009.

49 Kotera, J., Smidl, V., Sroubek, F.: 'Blind deconvolution with model discrepancies', *IEEE Transactions on Image Processing*, 2017, **26**, (5), pp. 2533–2544

50 Cai, J.F., Ji, H., Liu, C., Shen, Z.: 'Framelet-based blind motion deblurring from a single image', *IEEE Transactions on Image Processing*, 2012, **21**, (2), pp. 562–572

51 Shan, Q., Jia, J., Agarwala, A.: 'High-quality motion deblurring from a single image', *ACM Transactions on Graphics*, 2008, **27**, (3), pp. 73:1–73:10. Available from: `http://www.cse.cuhk.edu.hk/leojia/programs/deblurring/deblurring.htm`

52 Gilles, J., Ferrante, N.B.: 'Open Turbulent Image Set (OTIS)', *Pattern Recognition Letters*, 2017, **86**, pp. 38–41